Digital Transformation for Systems Engineering

MBSE Symposium

Co-chair: Maged Elaasar (JPL)
Co-chair: Sebastien Gerard (CEA-LIST)
January 24, 2019

Jet Propulsion Laboratory
California Institute of Technology

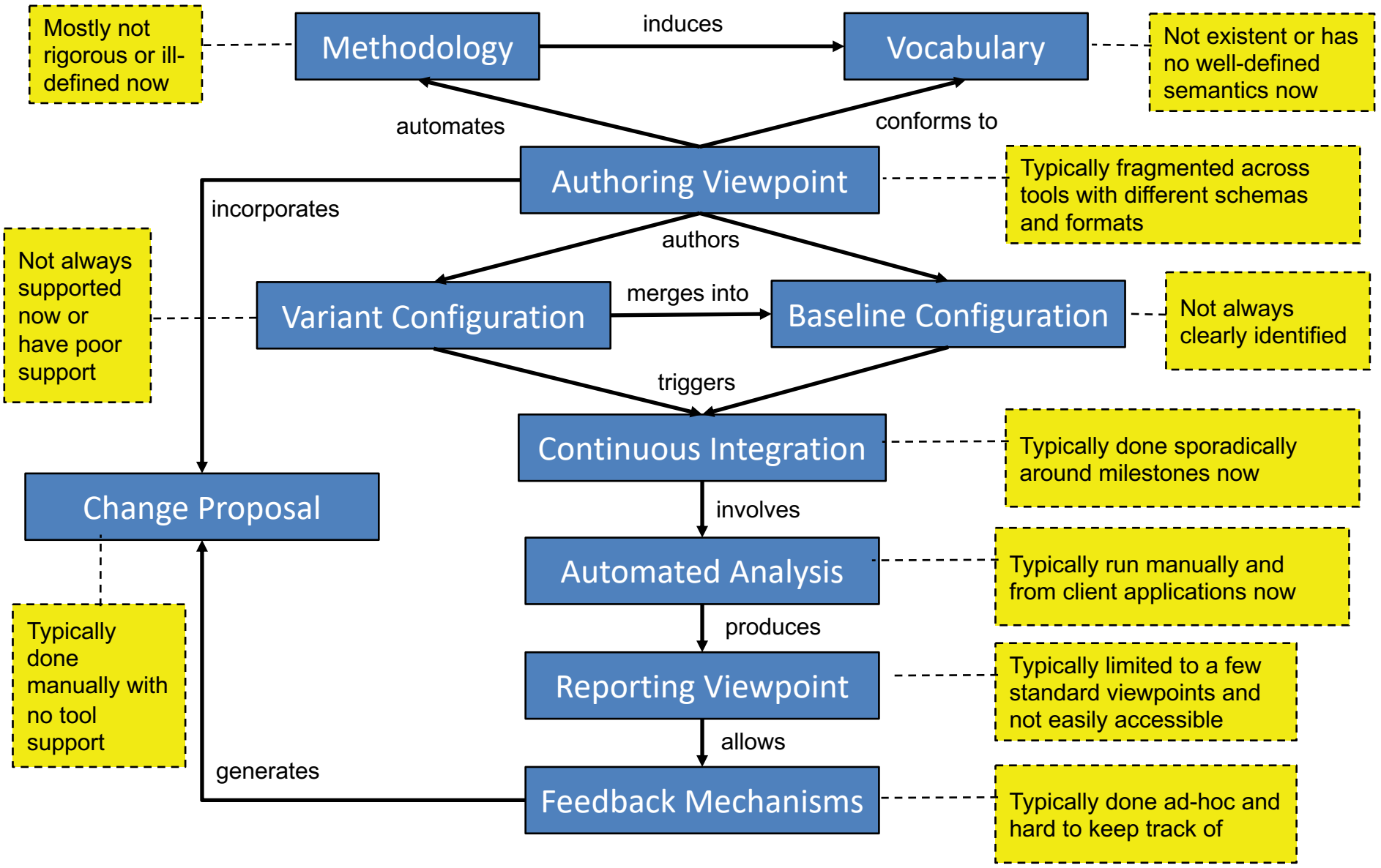
Disclaimer

- Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology

2h Agenda

- **Introduction** -5 mins
- **Open CAESAR Initiative**, Maged Elaasar (JPL) -45 mins
- **Papyrus Project**, Sebastien Gerard (CEA-LIST) -10 mins
- **Capella Project**, Stephane Bonnet (Thales) -10 mins
- **Sirius Project**, Stephane Lacrampe (Obeo) -10 mins
- **RangeDB Project**, Harald Eisenmann (Airbus) -10 mins
- **Discussion** -30 mins

Modern System Development Process



INTRODUCING THE

Open CAESAR Initiative

MBSE Symposium

Maged Elaasar, PhD
IMCE Program Chief Architect
January 24, 2019

Jet Propulsion Laboratory
California Institute of Technology

Key MBSE challenges

Challenges	Mitigations
System modeling **tools are too complex and rigid** to use for systems engineers	We need to use **domain-specific tools** to lower the bar to entry and provide **curated libraries and model templates**
Different projects **create system models differently** making it hard to reuse or analyze consistently	We need to define **modeling methodologies** and facilitate conforming to them using tools
Heterogeneous tools are used in SE because many discipline-specific analyses are needed	We need effective **tool integration** strategies, facilitating cross tool linking or interchange
Model content is **rapidly changing** as work progresses, increasing risk, and latency in decision making	We need effective cross-tools **configuration management**, **change management**, and **continuous automated analysis**
Models are being authored by **separate teams** with different concerns/expertise	We need to enable **federated multi-viewpoint architecture** with methods to keep the viewpoints aligned
Standard model **notation is not sufficient** to present their content and communicate it to stakeholders	We need ability to design **custom reporting viewpoints**, automatically generate and make them accessible

What is CAESAR?

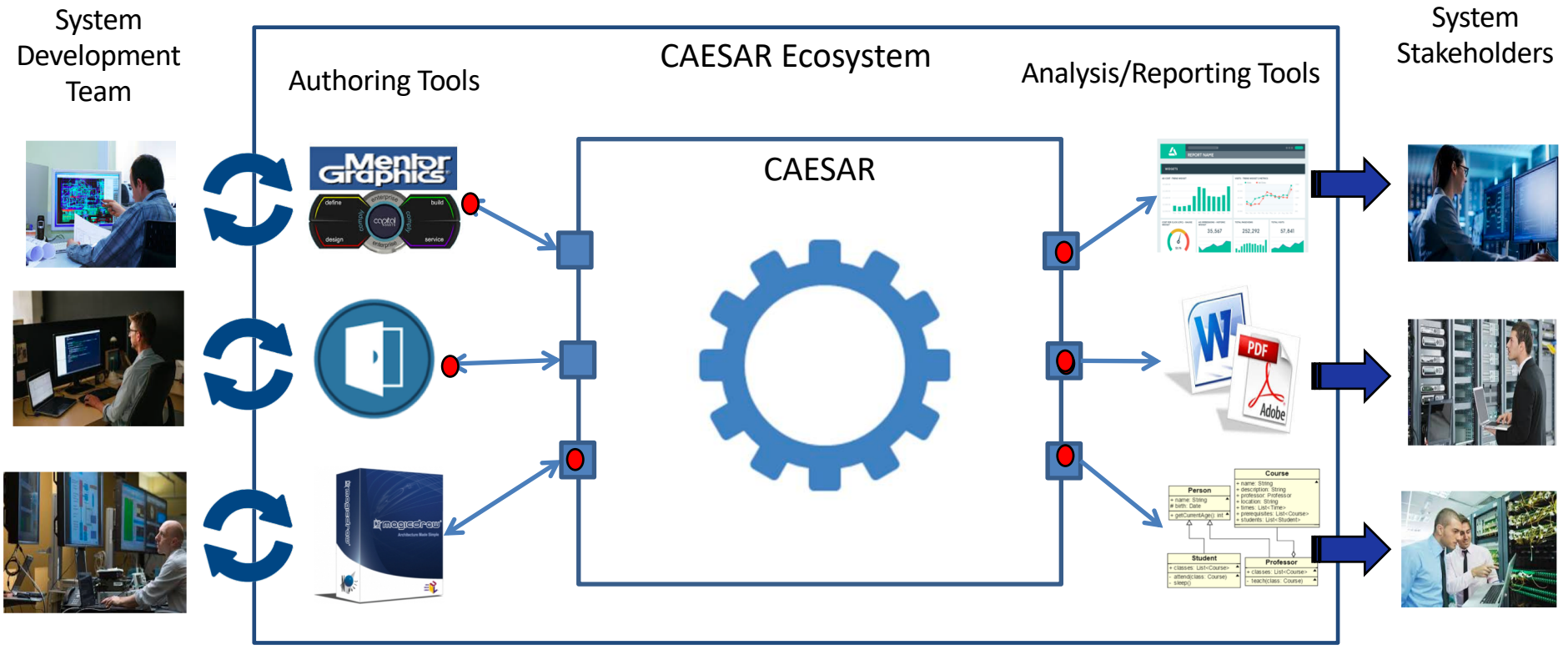
- CAESAR stands for **C**omputer **A**ided **E**ngineering for **S**ystems **AR**chitecture
  - A software platform to enable an integrated model centric approach to SE
    - Enables a methodology-driven use of SE tools to perform SE functions
    - Supports a federated multi-viewpoint agile systems development process
    - Promotes reuse of existing technology while minimizing vendor lock-in
    - Addresses key challenges of the current practice of MBSE
    - Infuses development best practices learned from software engineering

Model Based Systems Engineering (MBSE)

Integrated Model Centric Engineering (IMCE)

The CAESAR Concept

- **A semantic data warehouse system for MBSE**
  - Acquires information from SE authoring tools acting as data sources
  - Curates, integrates, configuration manages SE information baseline in one place
  - Verifies the consistency of federated information and measures precise differences
  - Analyzes information, generates reports and proposes changes



Key Functions

Information Authoring

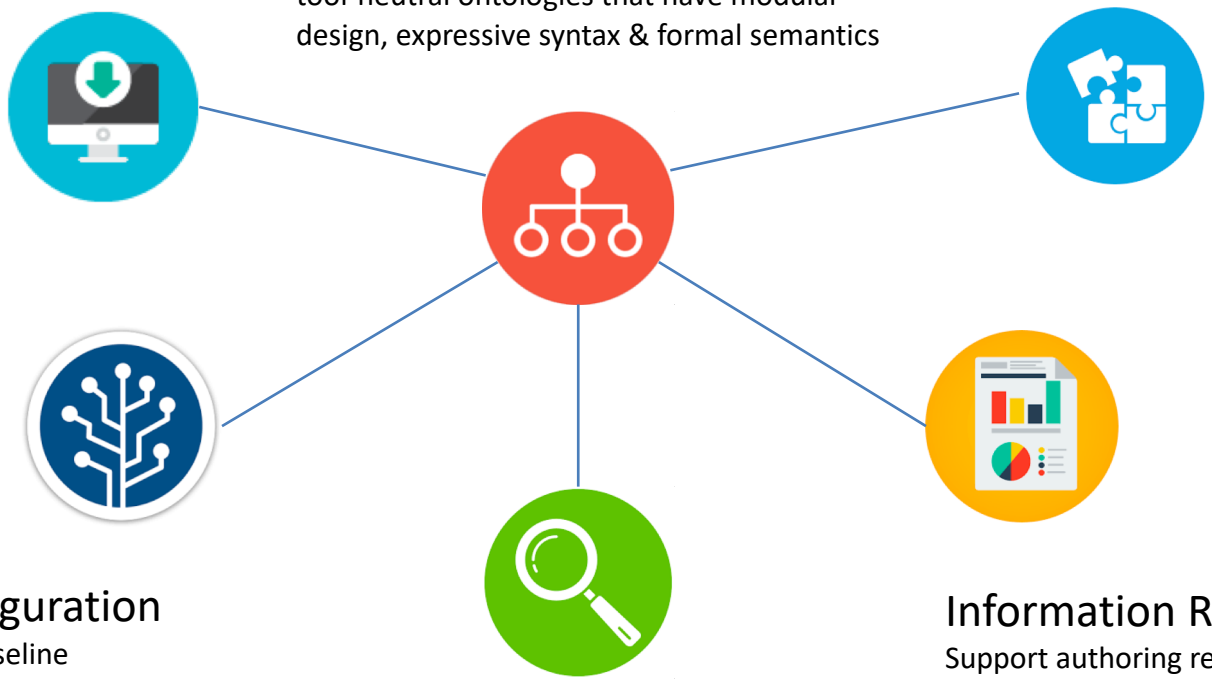
Supports authoring information by different tools, which can be existing tools that have been adapted or new methodology-specific tools

Information Representation

Supports representing information using tool-neutral ontologies that have modular design, expressive syntax & formal semantics

Information Integration

Supports defining methodology specific integrating workflows for the information fragments and proposing changes to keep them consistent



Information Configuration

Supports maintaining a baseline configuration for information (as well as change proposal configurations) and version controlling the results or running integration workflows

Information Analysis

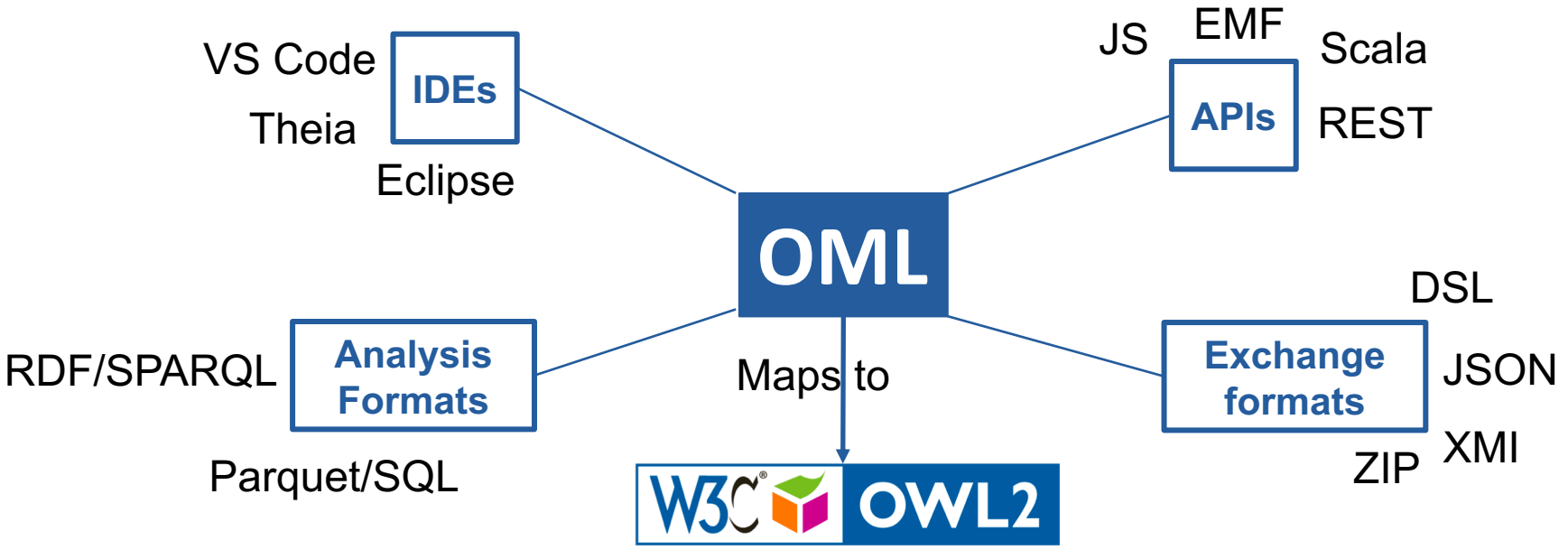
Support analyzing information with description logic reasoners and by other analysis tools using published query end points and APIs

Information Reporting

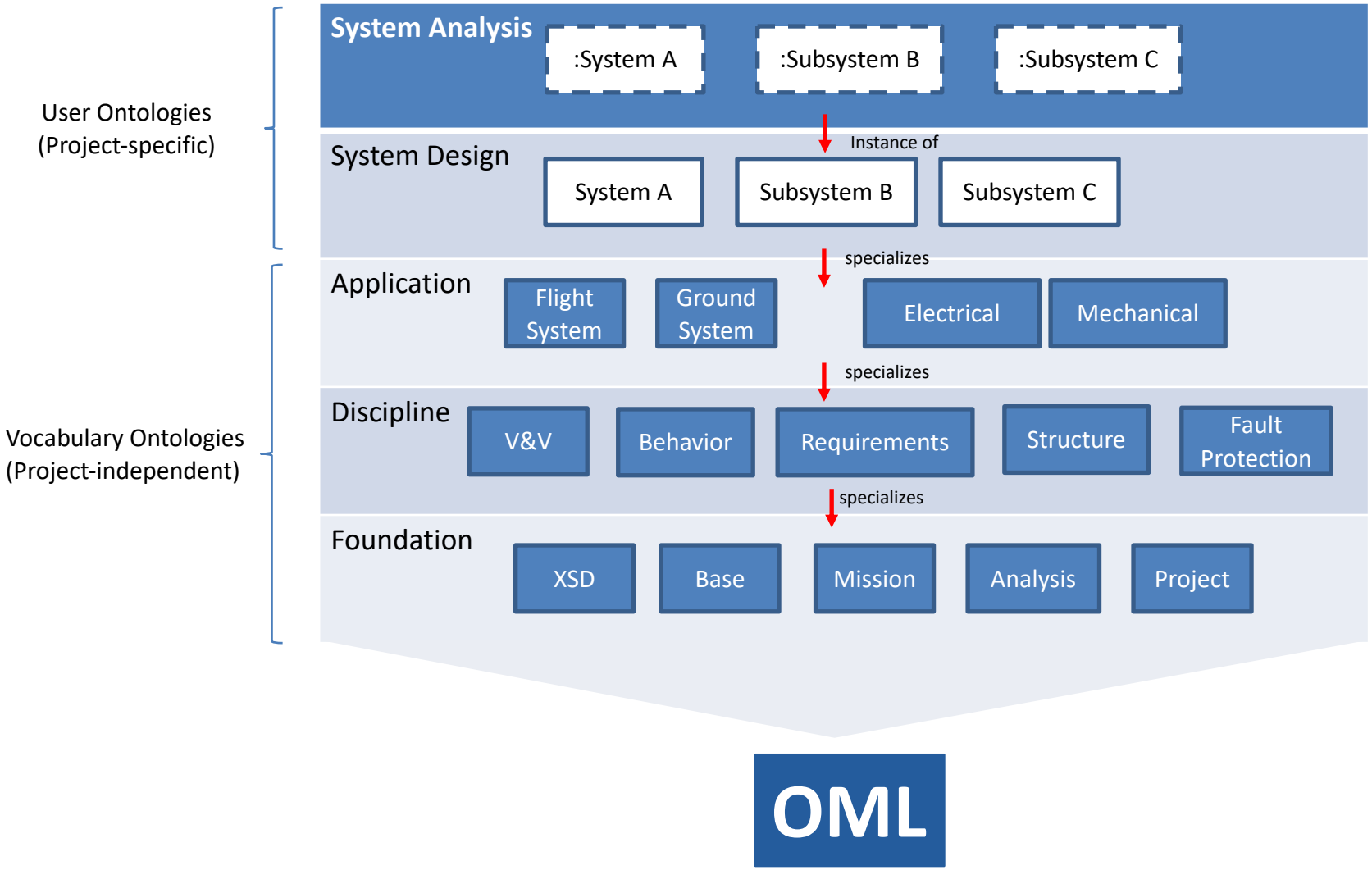
Support authoring reports on information to address different stakeholder concerns, regenerating them when the integration workflows rerun and organizing them in accessible reporting dashboards

Information representation

- CAESAR represents Information as a set of ontologies expressed in the **Ontology Modeling Language** (OML)
- OML maps to the W3C standard **Web Ontology Language 2** (OWL2)
- OML has expressive syntax, formal (**Description Logic**) semantics, and several sets of APIs, IDEs, exchange formats, and analysis formats
- OML is a tool-neutral representation that allows representing information, exchanging it, and analyzing it using a variety of tools



Information architecture

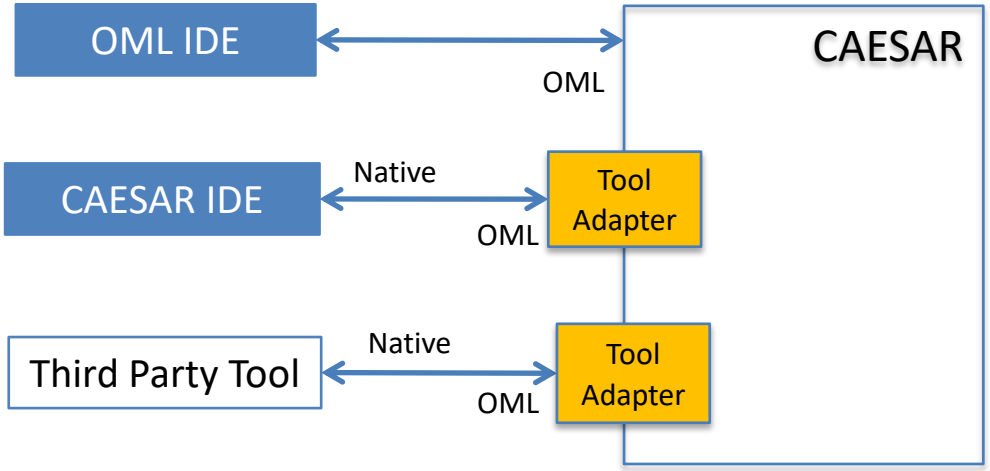


Information authoring

- CAESAR enables authoring ontologies directly in OML textual/graphical DSLs using one of its **OML IDEs** (Eclipse, VS Code, and Theia)

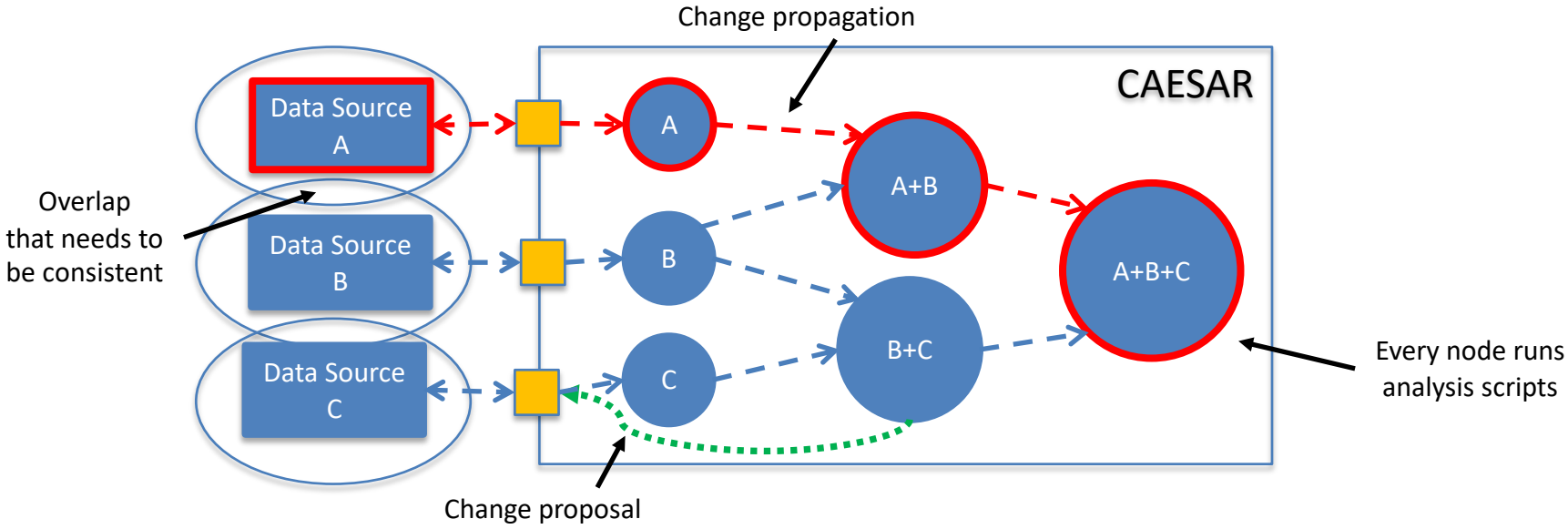
- CAESAR enables authoring ontologies using its own **CAESAR IDE**, a multi-disciplinary modeling tool which can be extended with custom viewpoints

- CAESAR also enables authoring ontologies using **third party tools** that have been adapted using a tool adapter (which supports bi-directional transformations between OML and the tool's native format)



Information integration

- CAESAR enables methodology-specific **incremental integration** of ontology fragments (from different tools) into a unified dataset
- The integration methodology is specified as a **workflow graph** whose nodes are configured with analysis scripts and that can be run manually or automatically based on events
- CAESAR helps maintain consistency between the integrated fragments by **proposing changes** to those fragments to restore consistency

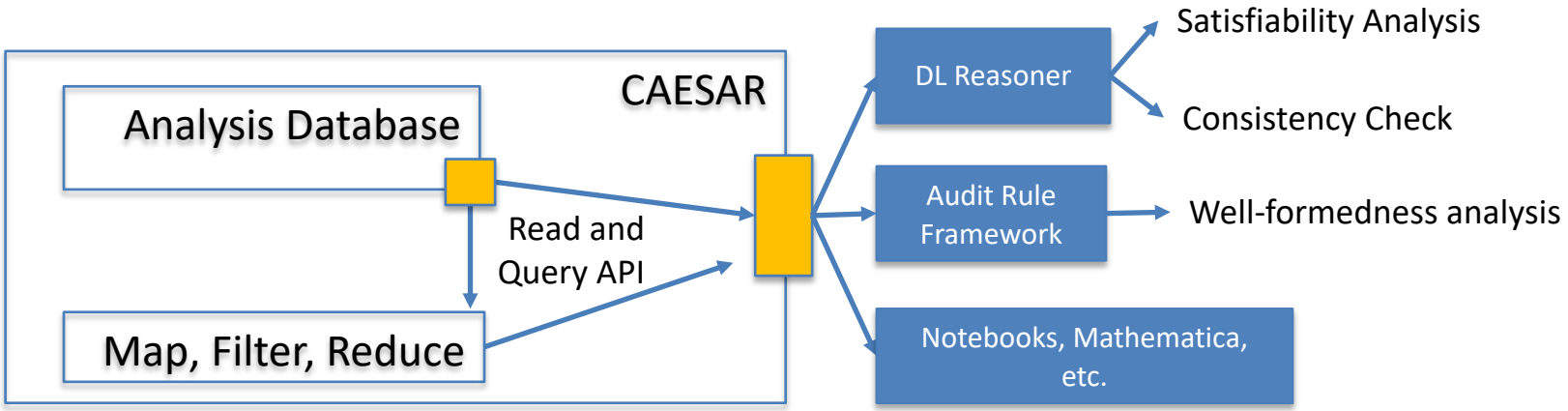


Information analysis

- CAESAR supports **read APIs**, a **rule language** and a **query language**, which can be used to perform powerful analysis on the dataset

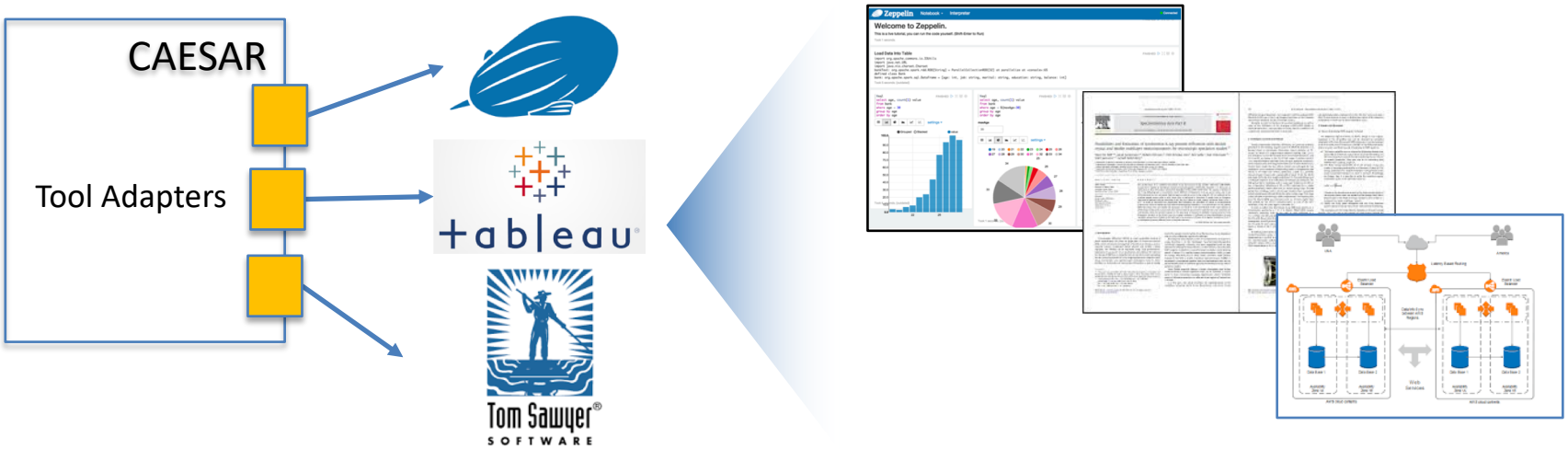
- CAESAR supports some default analyses through a DL reasoner like **satisfiability analysis** (every class can be instantiated) and **consistency checking** (no contradictory axioms exist)

- CAESAR supports analysis via **third party analysis tools** by transforming the dataset to other formats expected by those tools



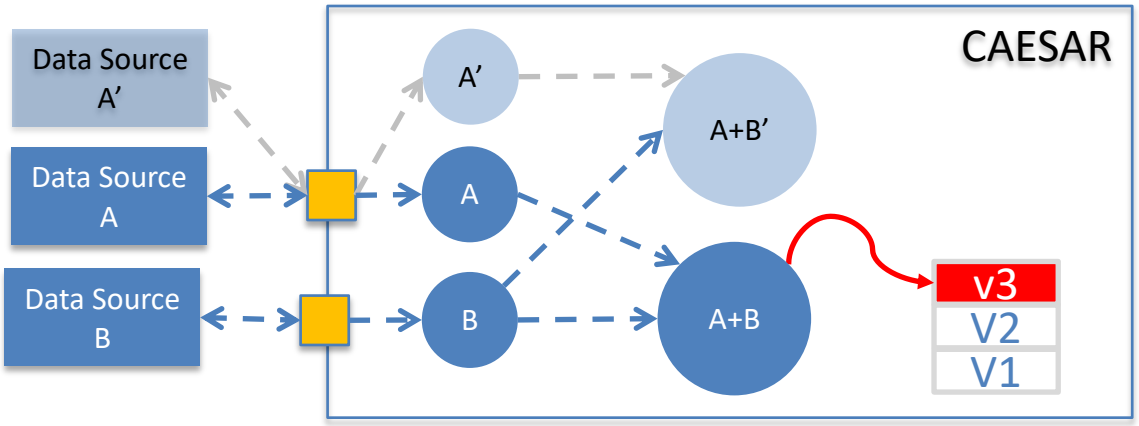
Information reporting

- CAESAR allows creating reports on datasets or their analysis results using a variety of **reporting tools** to address stakeholders concerns
- Reports can be automatically **produced during integration** and stored for efficient viewing later (historical reports are archived)
- Reports are **published on a project portal**, where they are organized in categories and can be searched and viewed (with access control)
- Reports can be **commented on** and/or **approved** by stakeholders

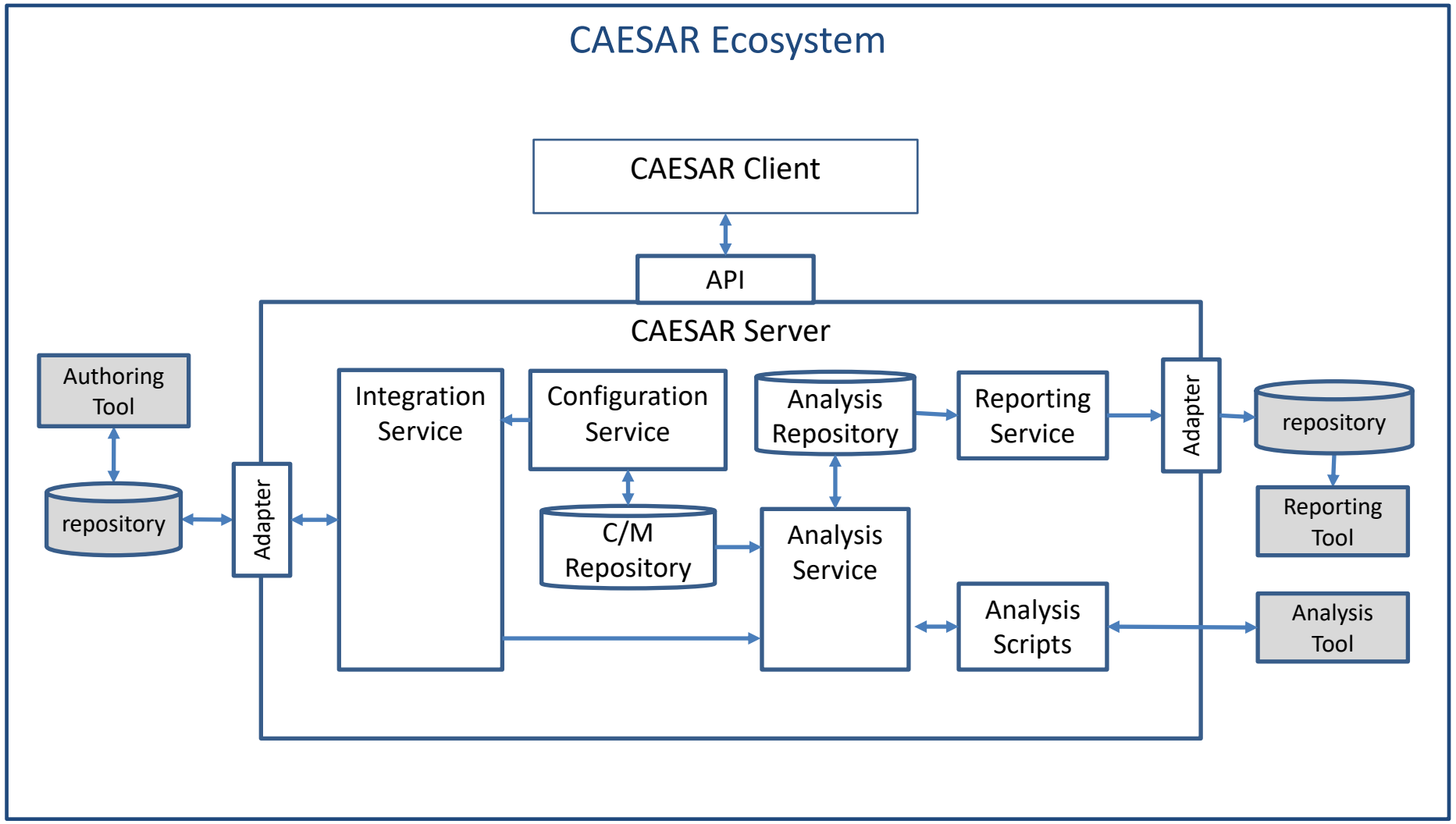


Information configuration

- CAESAR allows a project to have a **baseline** configuration of data sources, as well as **variant** configurations (representing change proposals)
- The project integration workflow can run on any configuration allowing the **impact of changes** to be analyzed before merging them into baseline
- The entire **history of integration** (including datasets, analysis results and reports) with full **provenance** is version controlled
- The configuration history enables **provenance audit**, **trend analysis**, as well as the possibility for **incremental analysis**



CAESAR conceptual architecture



CAESAR@JPL

Systems Engineering & Formulation
Division 31

Engineering and Science
Directorate 3x

hosts

Engineering Development Office
Program Management Office

sponsors

manages

Software Architect

Leads

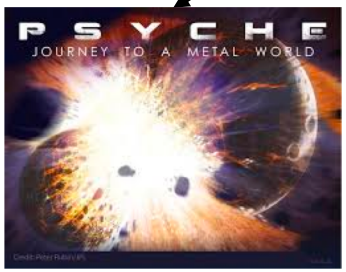
Integrated Model Centric Engineering
R&D Program

develops

CAESAR

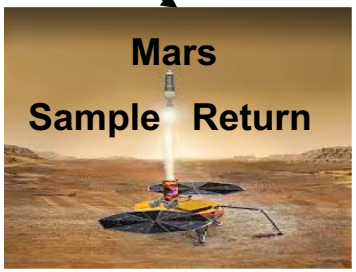
Hydrogen release FY18

supports

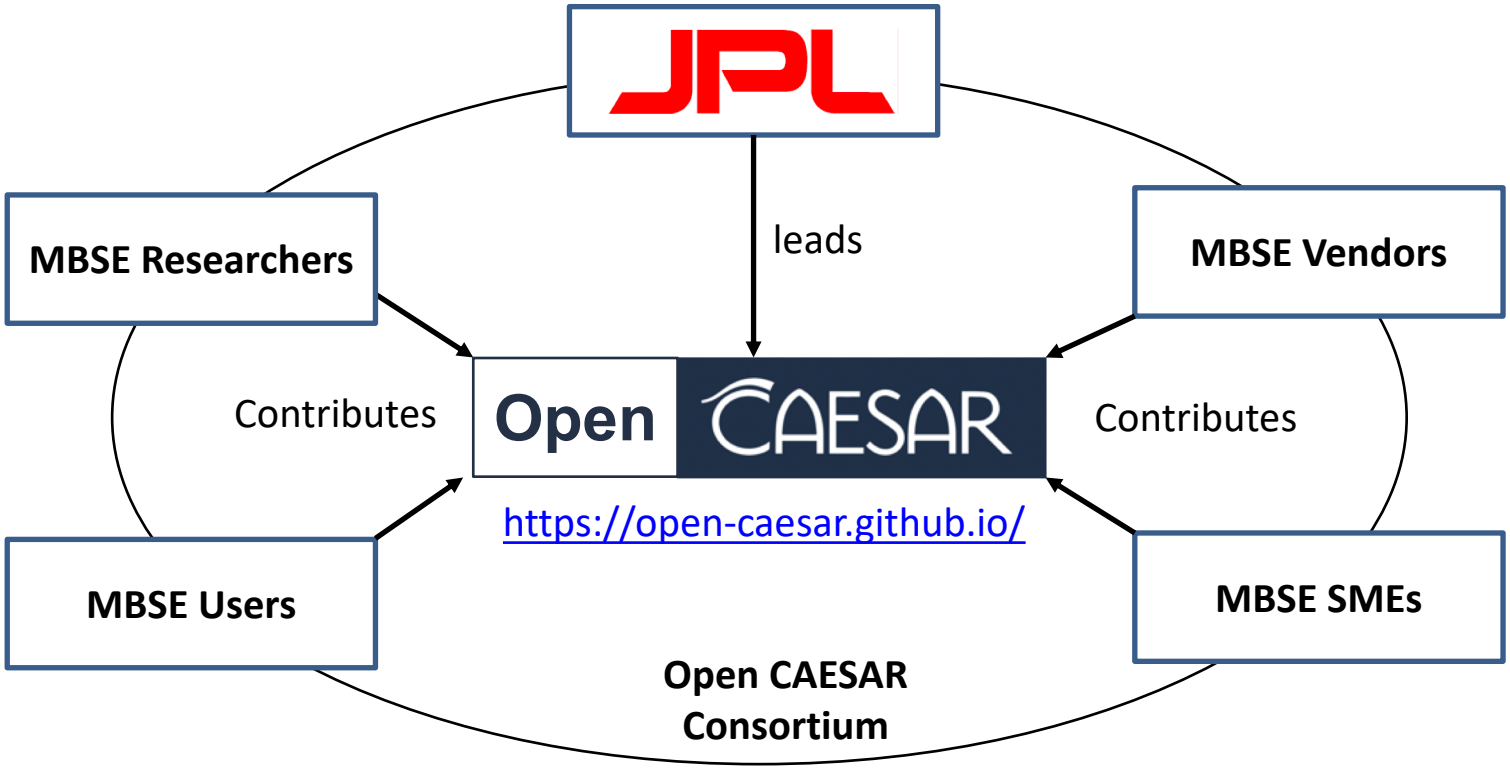


...

...



Open CAESAR Initiative



Open CAESAR: possible contributions

- **Information representation**
  - OML language definition and tooling
- **Information architecture**
  - Vocabularies for SE disciplines / applications
- **Information authoring**
  - Authoring tools and their adapter services
- **Information integration**
  - Infrastructure for integrating / synchronizing data sources
- **Information analysis**
  - Analysis tools and their adapter services
- **Information reporting**
  - Reporting tools and their adapter services
- **Information configuration**
  - Infrastructure for variability, change proposal, provenance
- **Example projects**
  - Case study projects to demonstrate ideas

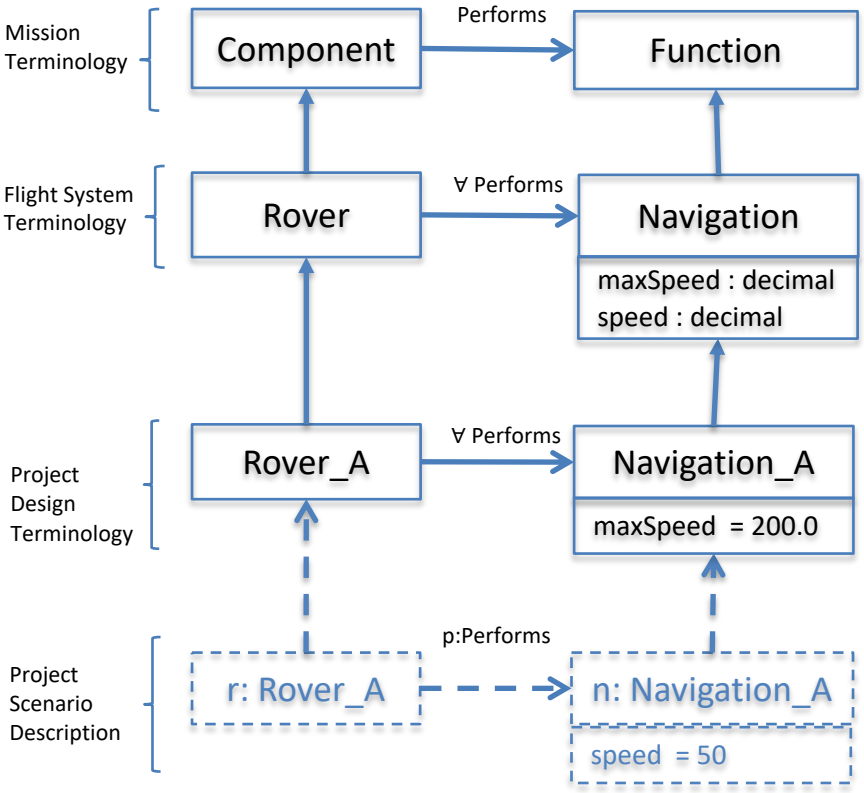
Open CAESAR: logistics

- **Create the consortium as an entity**
  - Define consortium charter, membership rules, and legal framework
  - Create an architecture council and appoint members
  - Create a program office and appoint members
  - Adopt an open source license (e.g., Apache 2.0)

- **Manage the open source code in GitHub.com**
  - Create GitHub organization for Open CAESAR
  - Create a separate repository for each major component
  - Create a GitHub Pages site (and wiki) for each component
  - Enable CI/CD for every repository
  - Assign a development team to lead each repository
  - Define work packages as GiHub organization projects

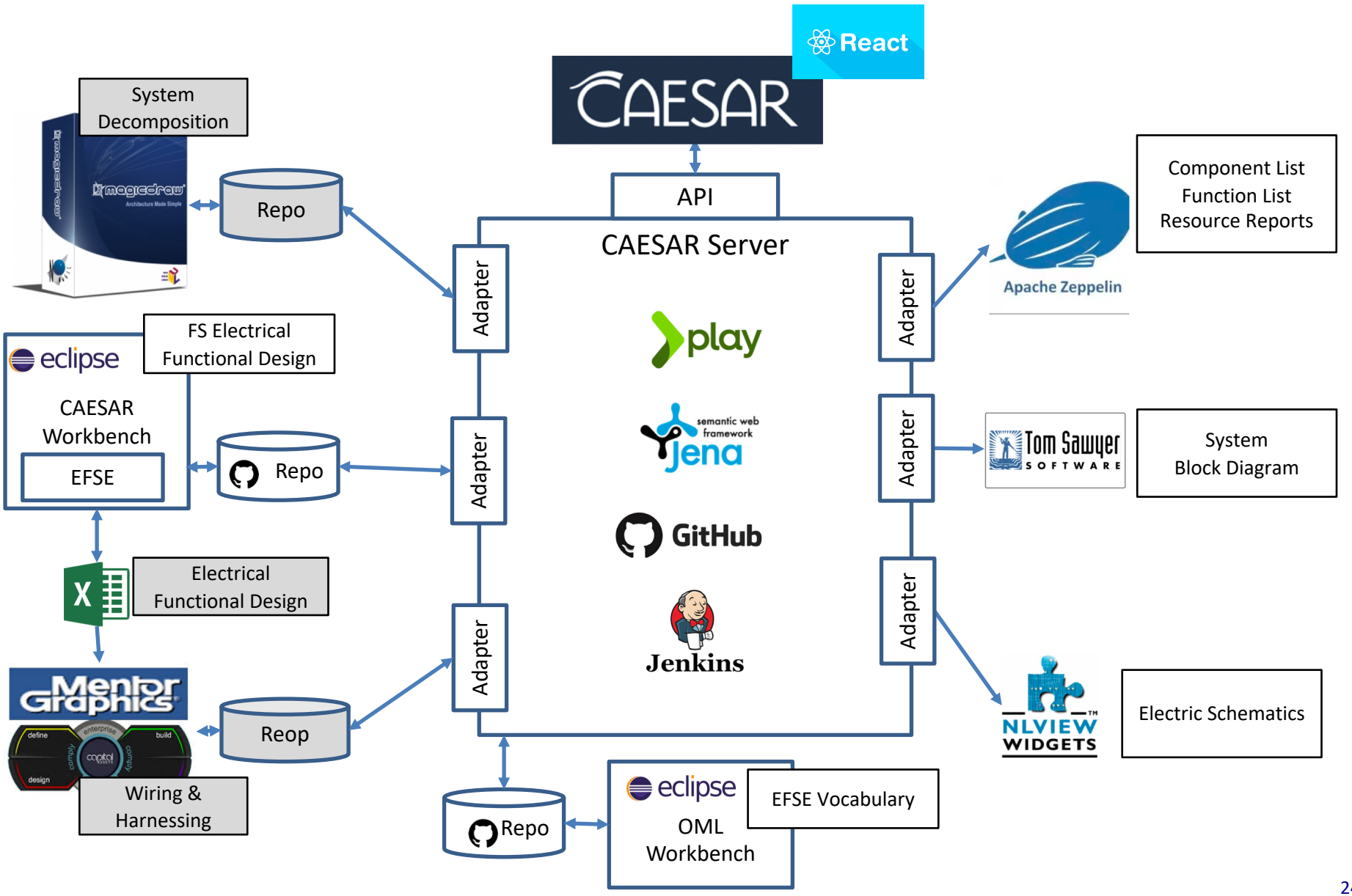
BACKUP

OML Example

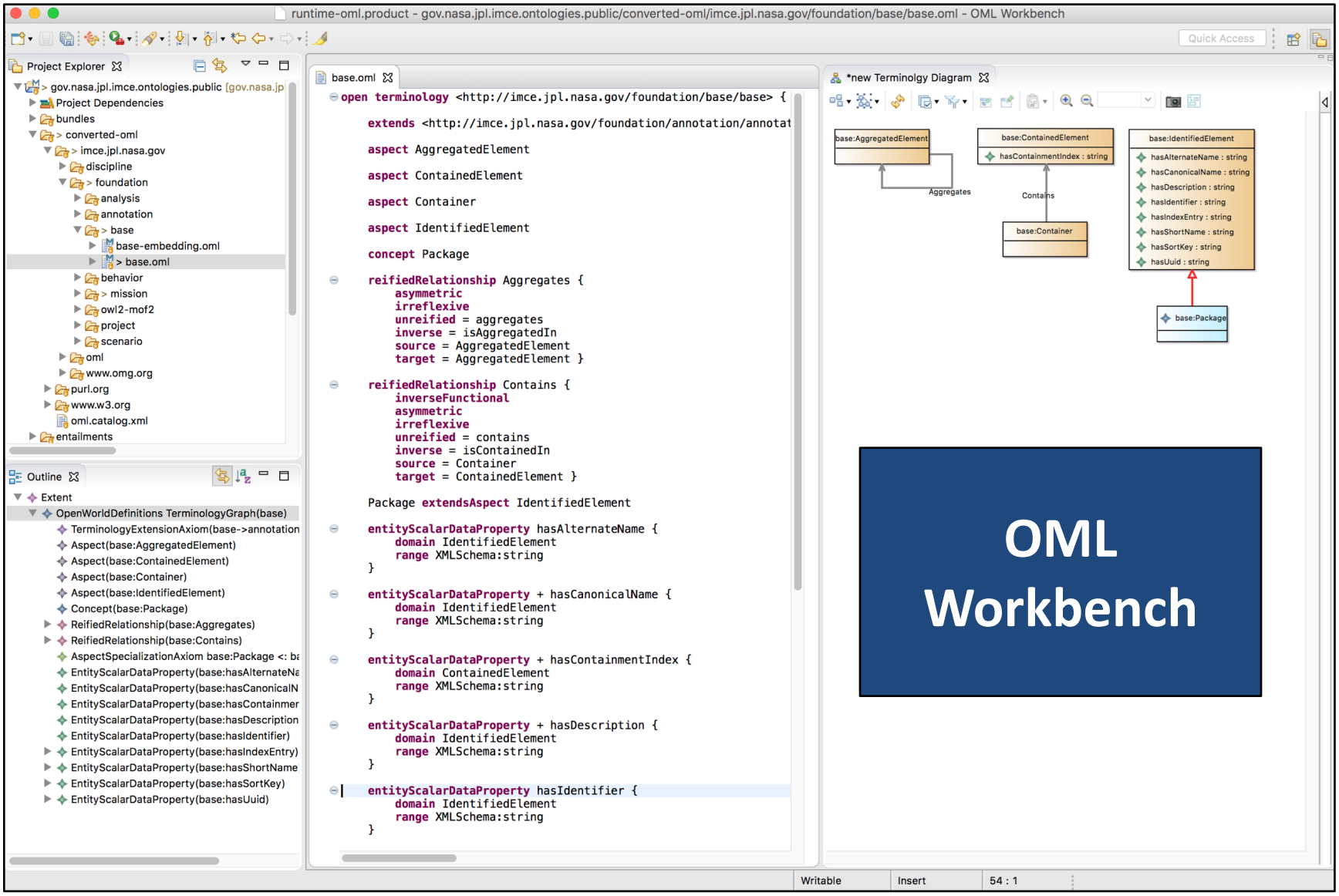



```
open terminology https://flightsystem {
   extends <https://mission>
   concept Rover {
      extends mission:Component
      restrictsAll mission:Performs to Navigation
   }
   concept Navigation {
      extends mission:Function
   }
   scalarDataProperty maxSpeed {
      domain Navigation
      range XMLSchema:decimal
   }
   scalarDataProperty speed {
      domain Navigation
      range XMLSchema:decimal
   }
}
closed terminology <https:/project/design> {
   extends <https://flightsystem>
   concept Rover_A {
      extends flightsystem:Rover
      restricsAll mission:Performs to Navigation_A
   }
   concept Navigation_A {
      extends flightsystem:Navigation
      restricts flightsystem:maxSpeed to 200.0
   }
}
closed description <https://project/scenario> {
   extends <https:///project/design>
   conceptInstance r : design:Rover_A
   conceptInstance n : design:Navigation_A {
      vzlueOf flightsystem:speed is 50
   }
   relationInstance p : mission:Performs {
      from r
      to n
   }
}
```

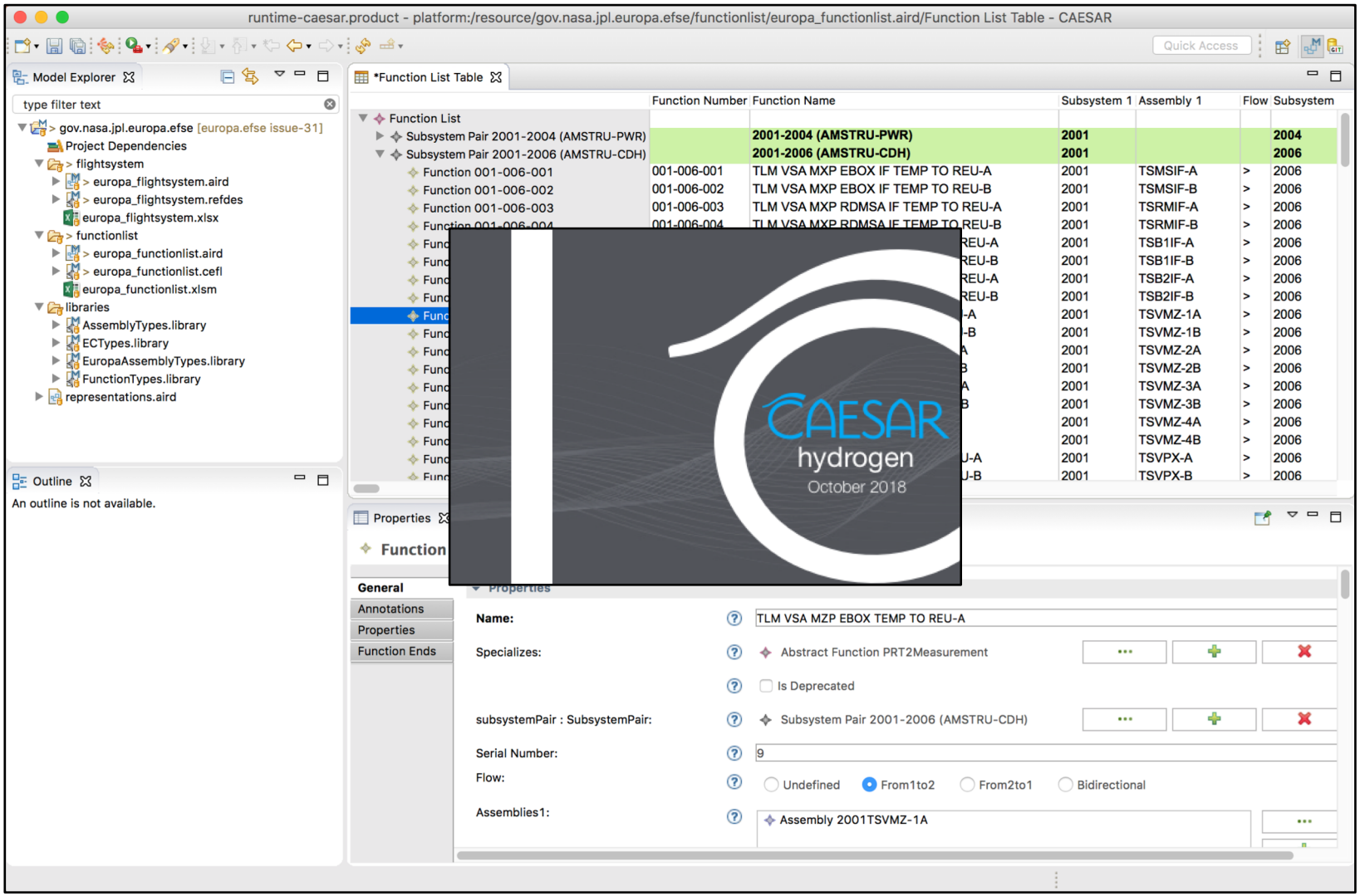
CAESAR Hydrogen release : EFSE Application



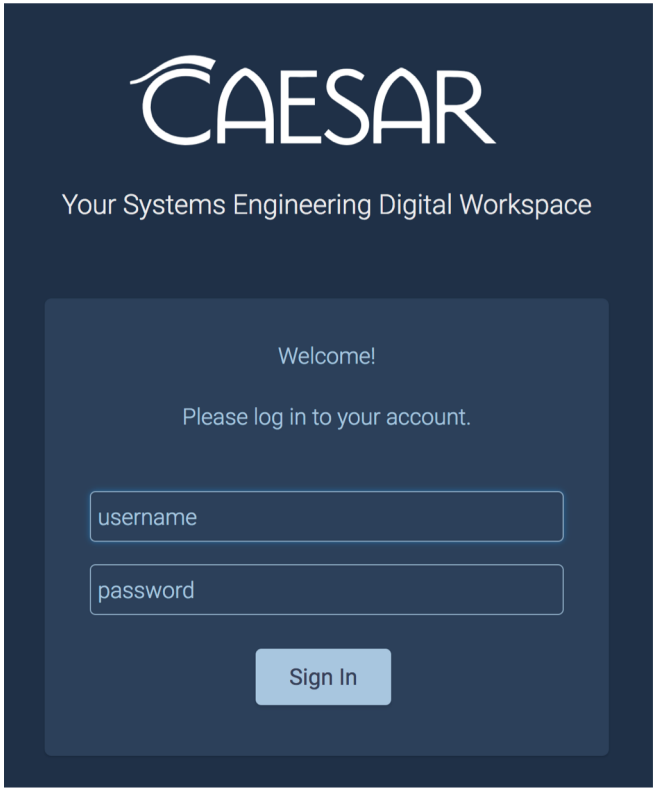
OML Workbench

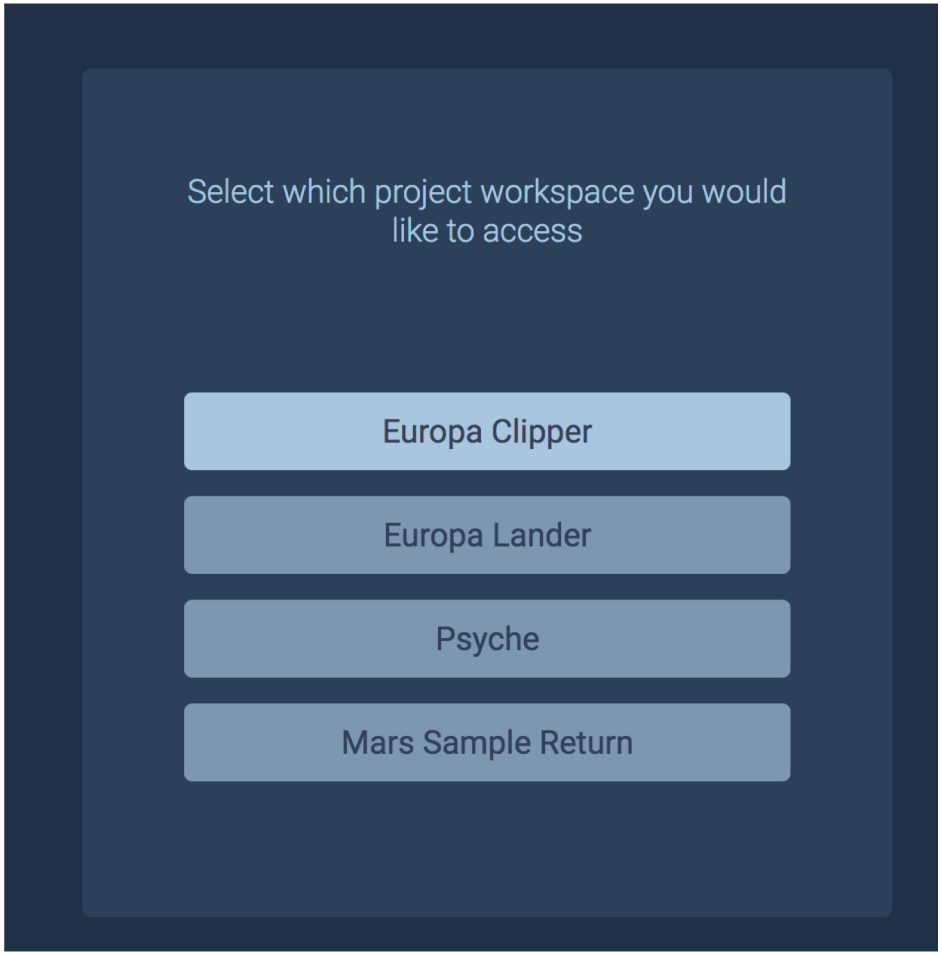


CAESAR Workbench: EFSE Application

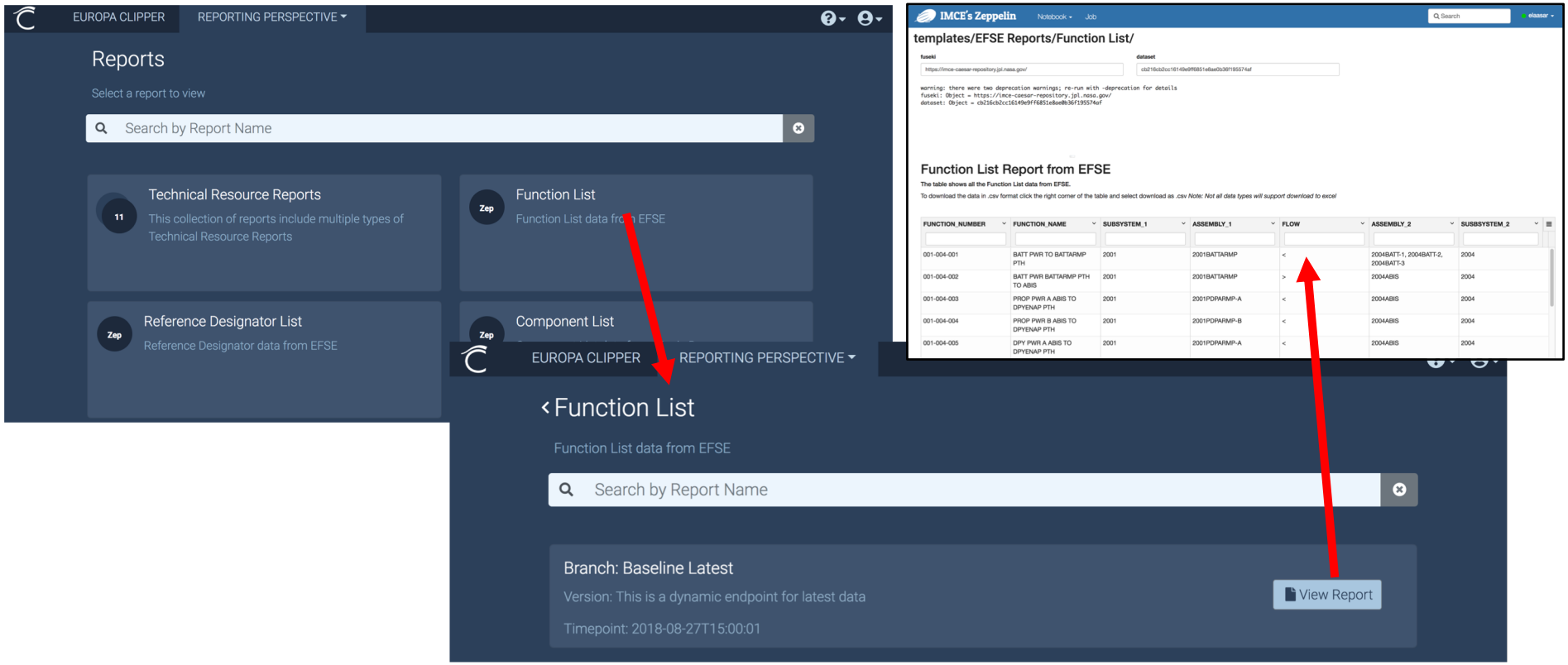


CAESAR Web App

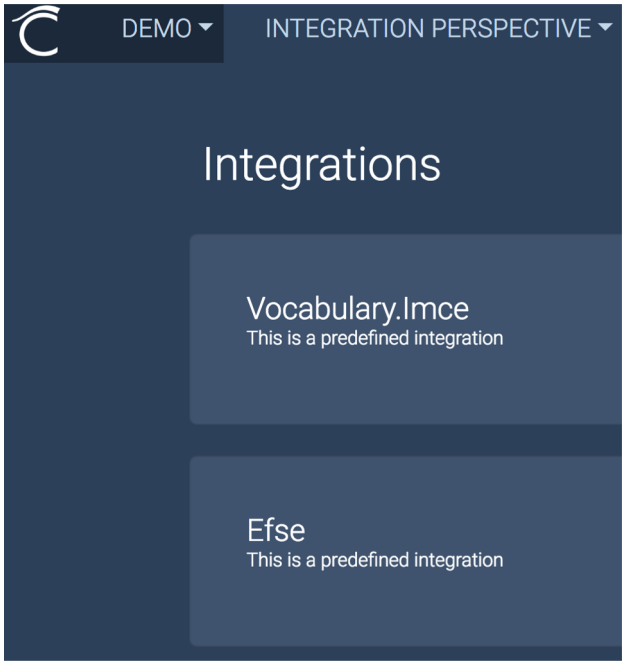


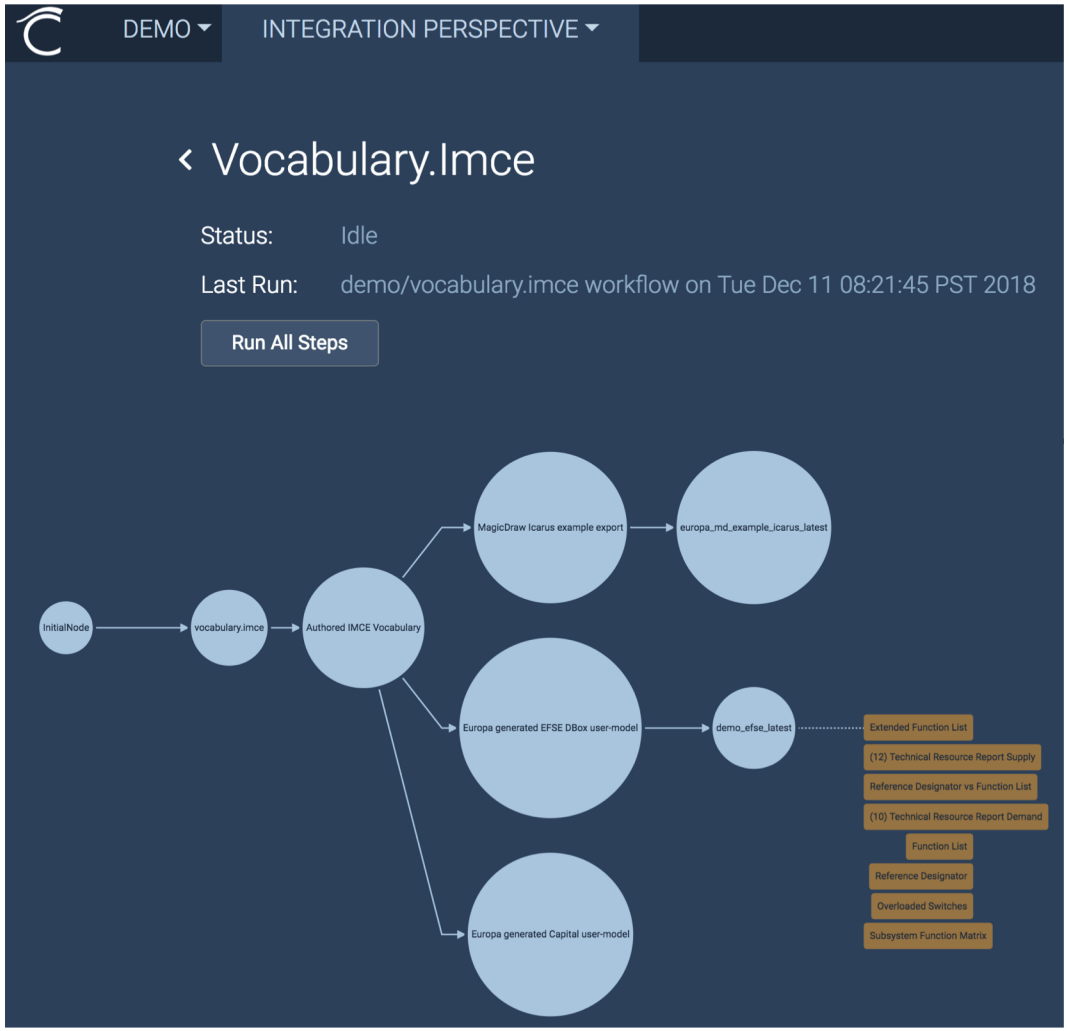


CAESAR Web App (Reporting Perspective)



CAESAR Web App (Integration Perspective)

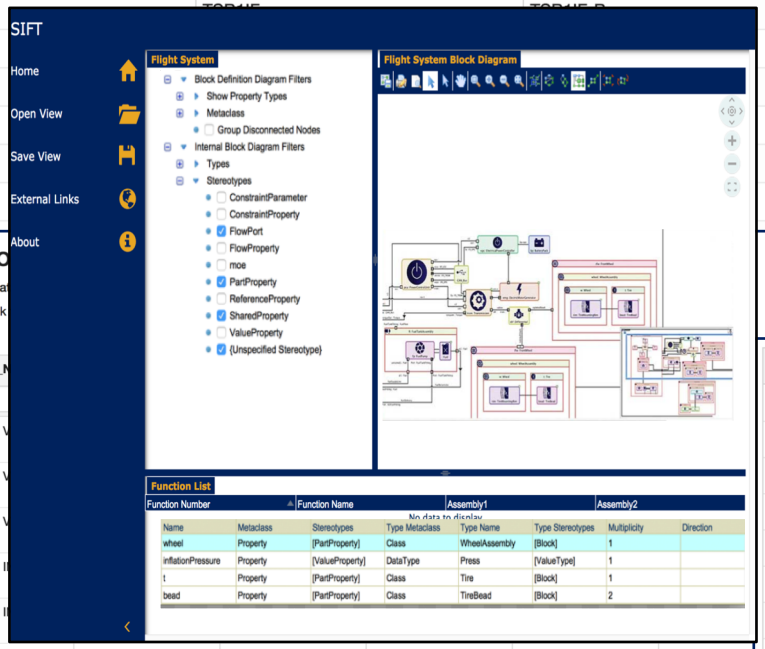




Example Reports

SUBSYSTEM_NUMBER	ACRONYM	NAME	REFERENCE_DESIGNATOR
2001	PDPARMP	PDPARMP-A	2001PDPARMP-A
2001	PDPARMP	PDPARMP-B	2001PDPARMP-B
2001	TSB1IF	TSB1IF-A	
2001			
2001			
2001			
2001			
2001			
2001			2AHIF-B

ss1_id	2000	2001	2002	2004	2005	2006	2007	2008	2009	2010	2011	2012	2021
2000	·	·	·	·	·	·	·	·	·	·	·	·	·
2001	·	·	·	8	·	70	·	·	·	·	·	·	·
2002	·	·	4	24	·	114	·	·	·	·	·	·	·
2004	·	·	·	9	10	28	52	2	16	·	26	10	60
2005	·	·	·	·	·	24	·	·	·	·	·	·	·
2006	·	·	·	·	·	192	96	48	18	·	46	34	2
2007	·	·	·	·	·	·	56	·	6	·	·	·	·
2008	·	·	·	·	·	·	·	·	·	·	·	·	·
2009	·	·	·	·	·	·	·	·	10	·	·	·	·
2010	·	·	·	·	·	·	·	·	·	56	·	·	120
2011	·	·	·	·	·	·	·	·	·	·	14	·	·
2012	·	·	·	·	·	·	·	·	·	·	·	28	·



Function List Repo

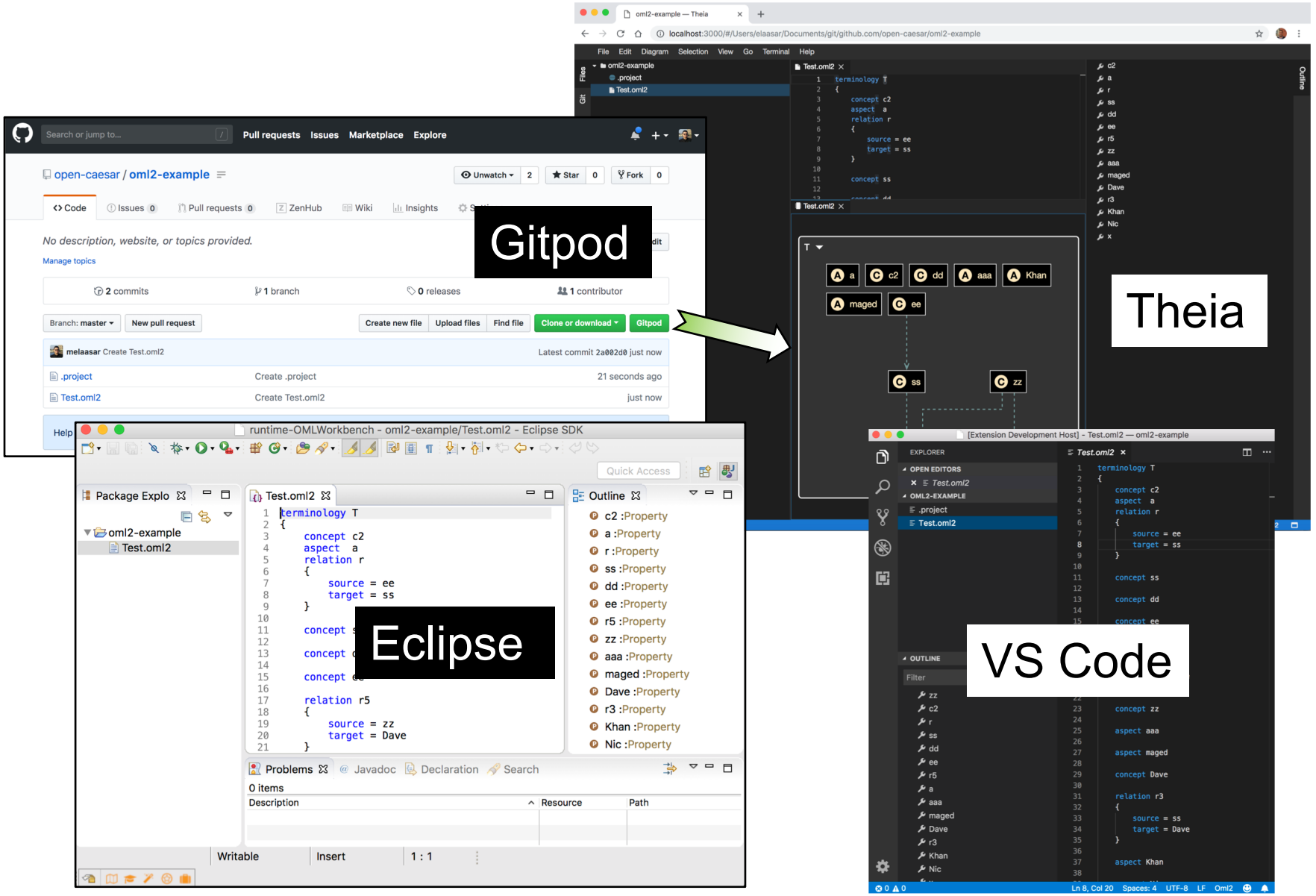
The table shows all the Function List da

To download the data in .csv format click

FUNCTION_NUMBER	FUNCTION_N					
004-004-001	TLM BATT-1 V PBC					
004-004-002	TLM BATT-2 V PBC					
004-004-003	TLM BATT-3 V PBC					
004-004-004	TLM BATT-1 I PBC PTH					
004-004-005	TLM BATT-2 I PBC PTH					
004-004-006	TLM BATT-3 INTEMP TO PBC PTH	2004	2004BATT-3	>	2004PBC	2004
004-004-007	TLM BATT-1 INTEMP TO PBC	2004	2004BATT-1	>	2004PBC	2004
004-004-008	TLM BATT-2 INTEMP TO PBC	2004	2004BATT-2	>	2004PBC	2004
004-004-009	TLM BATT-3 INTEMP TO PBC	2004	2004BATT-3	>	2004PBC	2004
004-004-010	UMB CMD BATT SW 1	2004	2004PBC	>	2004ABIS	2004

SUBSYSTEM	TOTALFUNCTION	2A	4A	5A
AMSTRU	79	6	0	0
CDH	958	0	0	4
EIS	94	13	0	1
ETHEMIS	21	2	0	1
GNC	210	4	0	4
ICEMAG	46	7	0	0
IMB	0	0	0	0
MASPEX	29	4	0	2
MISE	31	6	0	1
PIMS	29	4	0	0
PME	488	2	0	0
PMHARN	4	0	0	0
PMSTRU	614	0	0	0
PROP	340	0	0	0

Open CAESAR: OML Language Server



Open CAESAR: information representation WPs

- ***Basic tooling*:**
  - OML Abstract syntax, Java API, XMI interchange format (EMF)
  - OML Textual syntax / language server (Xtext)
  - OML Graphical syntax (Sprotty)
  - OML Semantics by mapping to OWL (Jena)
- ***Other APIs:***
  - OML Scala functional API
  - OML Java script API
  - OML REST API
- ***Other interchange formats:***
  - OML JSON format
  - OML Parquet format
  - OML Zip format

Open CAESAR: information architecture WPs

- ***Foundation vocabularies:***
  - Vocabulary for architecture description
  - Vocabulary for architecture integration
  - Vocabulary for diagram representation
- ***SE discipline vocabularies:***
  - Vocabulary for requirements
  - Vocabulary for use cases
  - Vocabulary for system structure
  - Vocabulary for system behavior
  - Vocabulary for V&V
  - Vocabulary for I&T
- ***SE application vocabularies:***
  - Vocabularies for different application domains (aerospace, automotive, etc.)

Open CAESAR: information authoring WPs

- ***OML IDEs:***
  - OML IDE support in Eclipse (textual editor, graphical viewer, EMF frameworks)
  - OML IDE support in Theia (textual editor, graphical viewer, Gitpod integration)
  - OML IDE support in VS Code (textual editor)
- ***EMF-based tools:***
  - New EMF-based tool for existing OML vocabulary
  - New OML vocabulary for an existing EMF-based tool
  - Generic OML adapter CLI based on OML-EMF vocabulary mapping
- ***UML-based tools:***
  - New UML-based tool for existing OML vocabulary
  - New OML vocabulary for an existing UML-based tool
  - Generic OML adapter CLI based on OML-UML vocabulary mapping
- ***Other tools:***
  - Framework for an OML adapter CLI
  - Template generator for an OML adapter CLI

Open CAESAR: information integration WPs

- ***Continuous integration and delivery:***
  - Vocabulary for describing a model integration workflow
  - Service to run a model integration workflow
  - Service to publish the results of a model integration flow
- ***Proposing change deltas to integrated SE tools:***
  - Way to represent change delta in OML
  - Framework to propose a change delta in OML
  - Framework to apply the change delta in a given tool

Open CAESAR: information analysis WPs

- ***Analysis on OML representation directly:***
  - Programmatic query API for OML
- ***Analysis using a triple store:***
  - Way to publish OML ontologies into a triple store
  - SPARQL endpoint for OML
  - Way to reason on OML with a DL reasoner
  - Higher level query language for OML
- ***Analysis using relational database:***
  - Way to publish OML ontologies into a relational database
  - SQL endpoint for OML
- ***Analysis using other analysis tools:***
  - Framework to analyze OML with other analysis tools
  - Framework for descriptive, predictive, prescriptive analytics

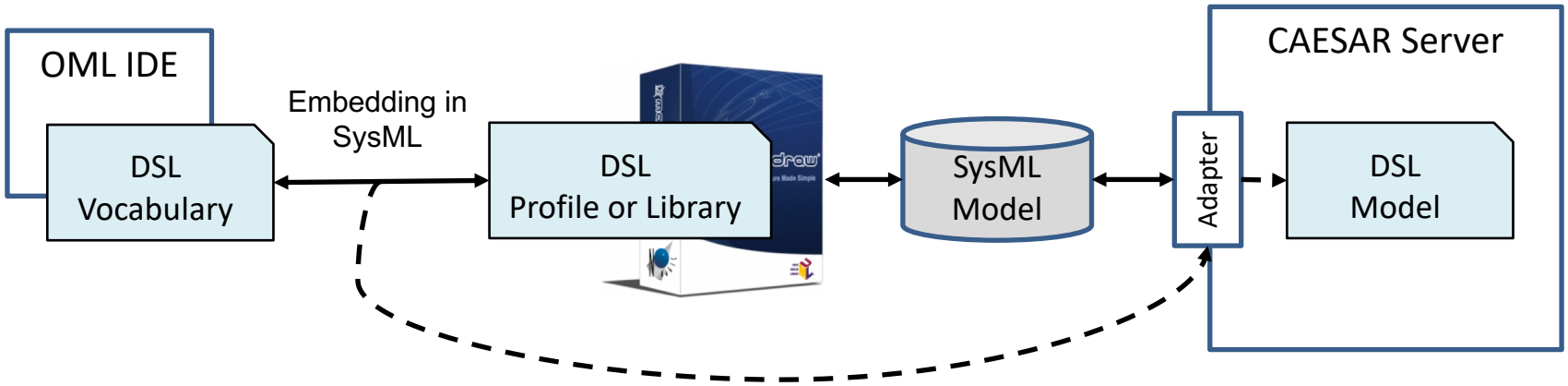
Open CAESAR: information reporting WPs

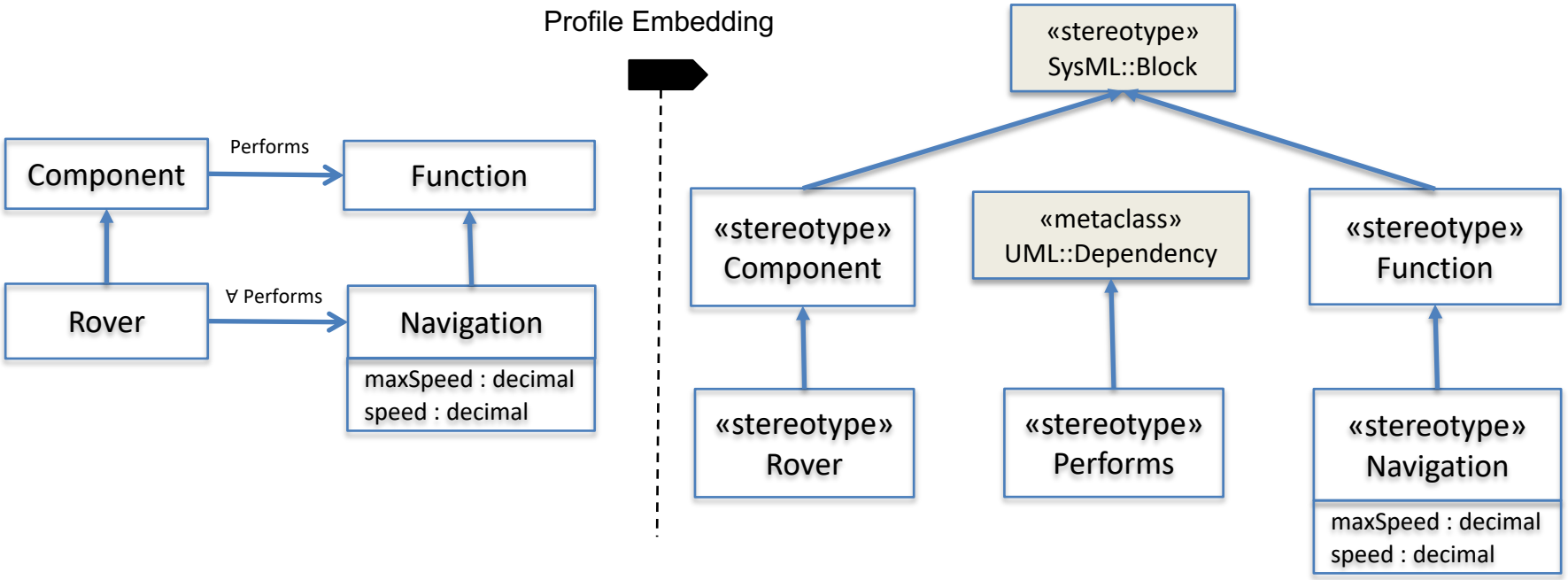
- ***Ontology visualization on the web:***
  - Web-based widgets for visualizing OML ontologies
- ***Ontology reporting tools:***
  - Interfaces for OML ontologies in various reporting tools
  - Way for reports to display change deltas
  - Way to report on trends in the evolution history
- ***Reporting dashboards:***
  - Way to organize reports to make it easy to browse/search
  - Way to review and comment OML ontologies
  - Way to approve changes to OML ontologies
  - Way to subscribe to changes in reports
- **Interactive Reporting tools**
  - Tool to browse the contents of the OML ontologies
  - Tool to assess impact of change to OML ontologies
  - Tool to run (elastic) text search on OML ontologies

Open CAESAR: information configuration WPs

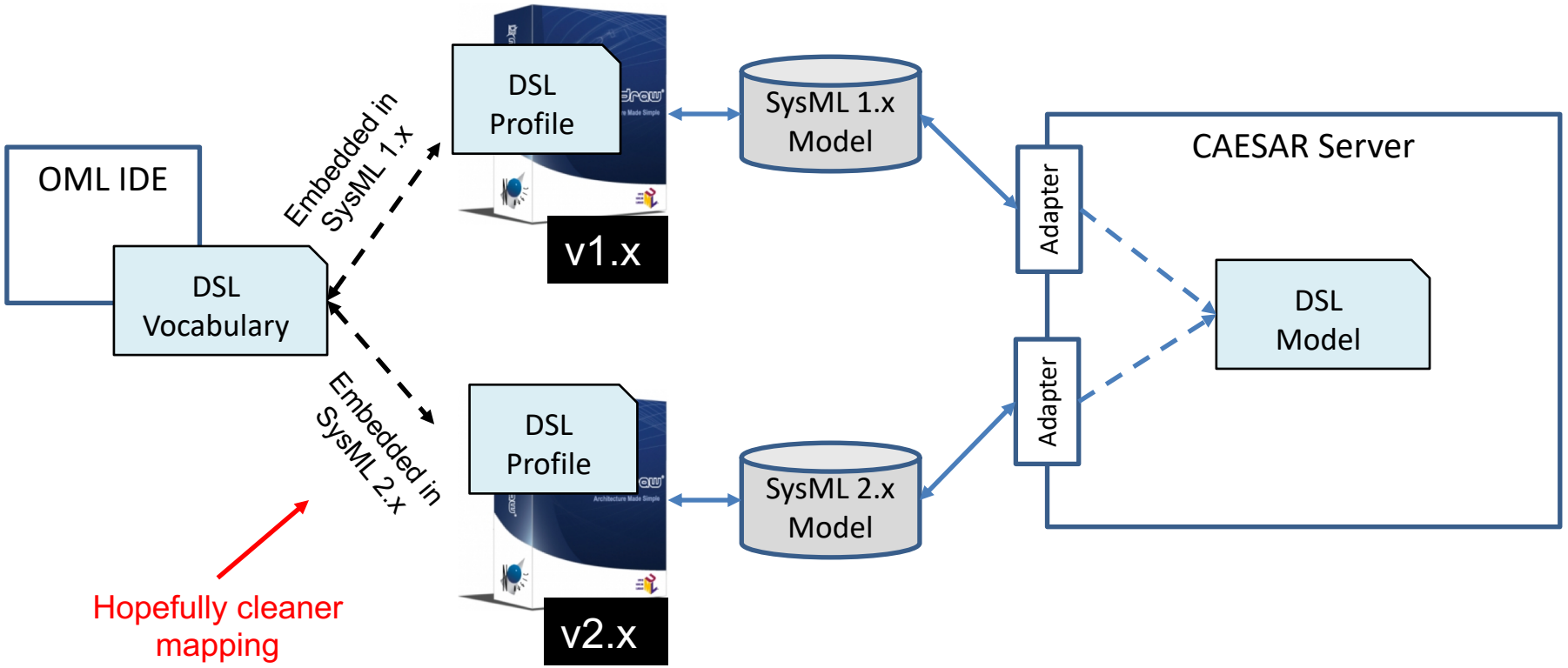
- ***Configuration management of ontologies:***
  - Way to manage OML ontologies in a SCM repository
  - Way to compare and merge ontologies in different repository configurations
  - Way to propose changes and doing pull requests using configurations

Open CAESAR SysML Adapter

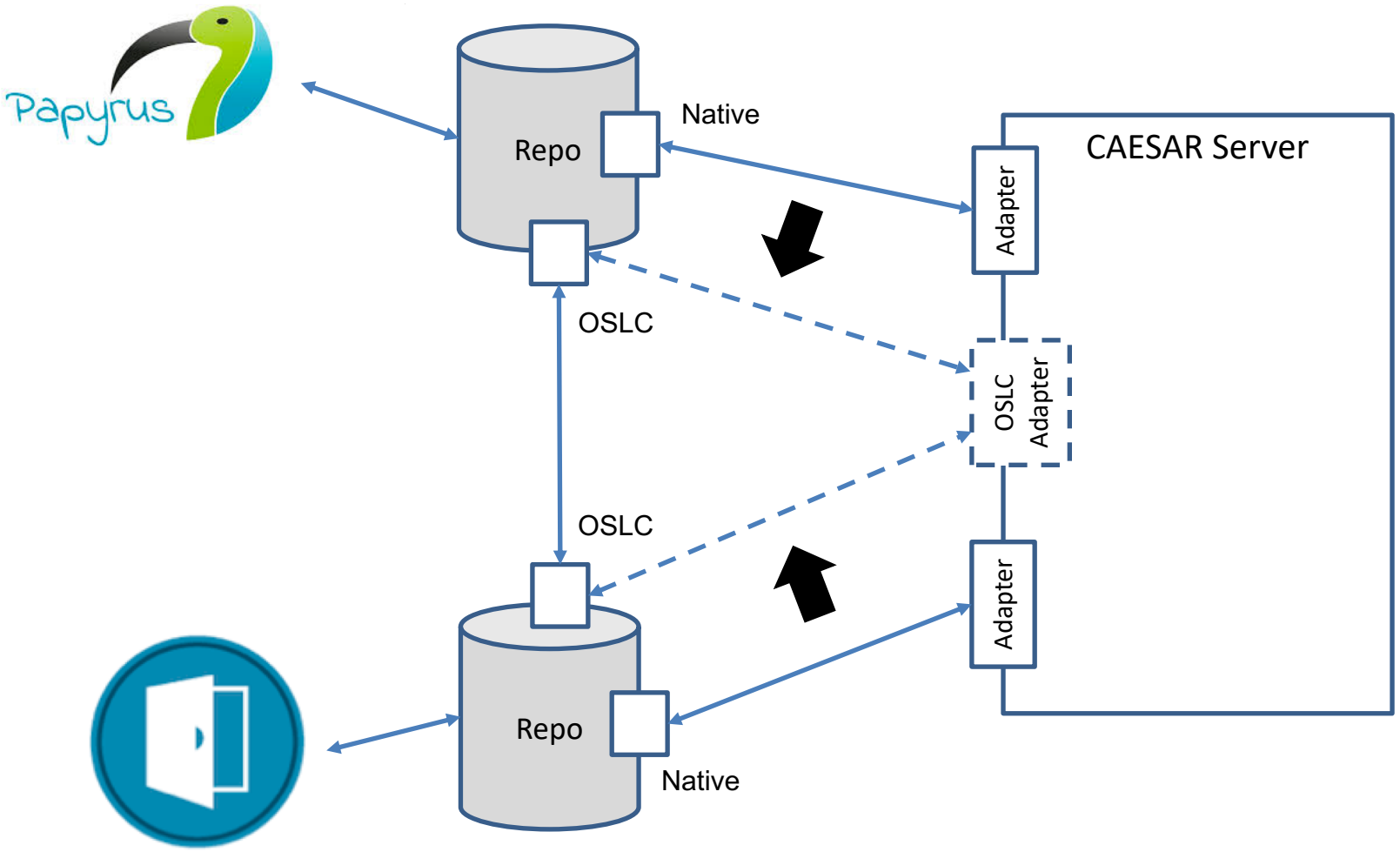




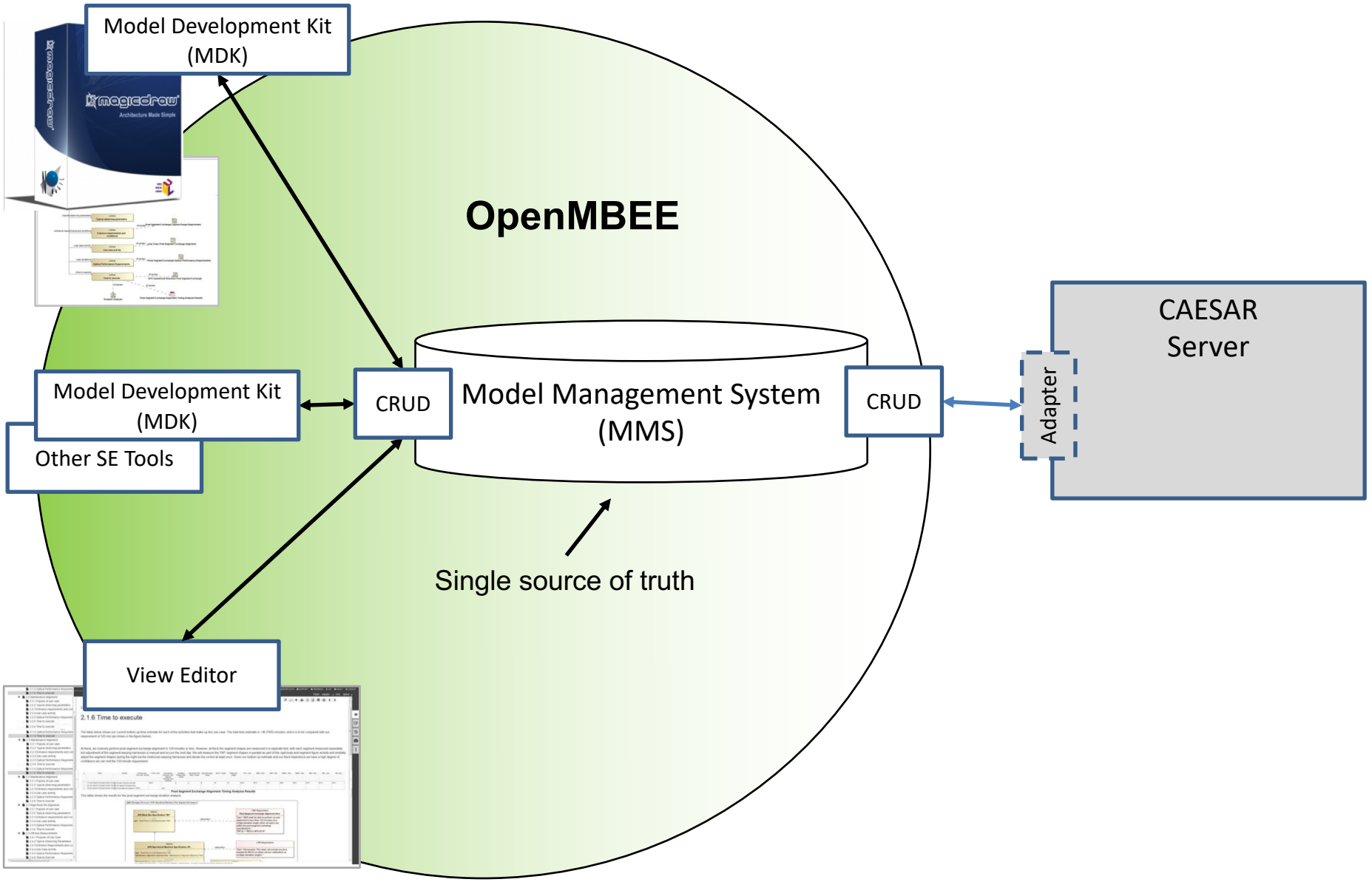
Open CAESAR vs. SysML v2



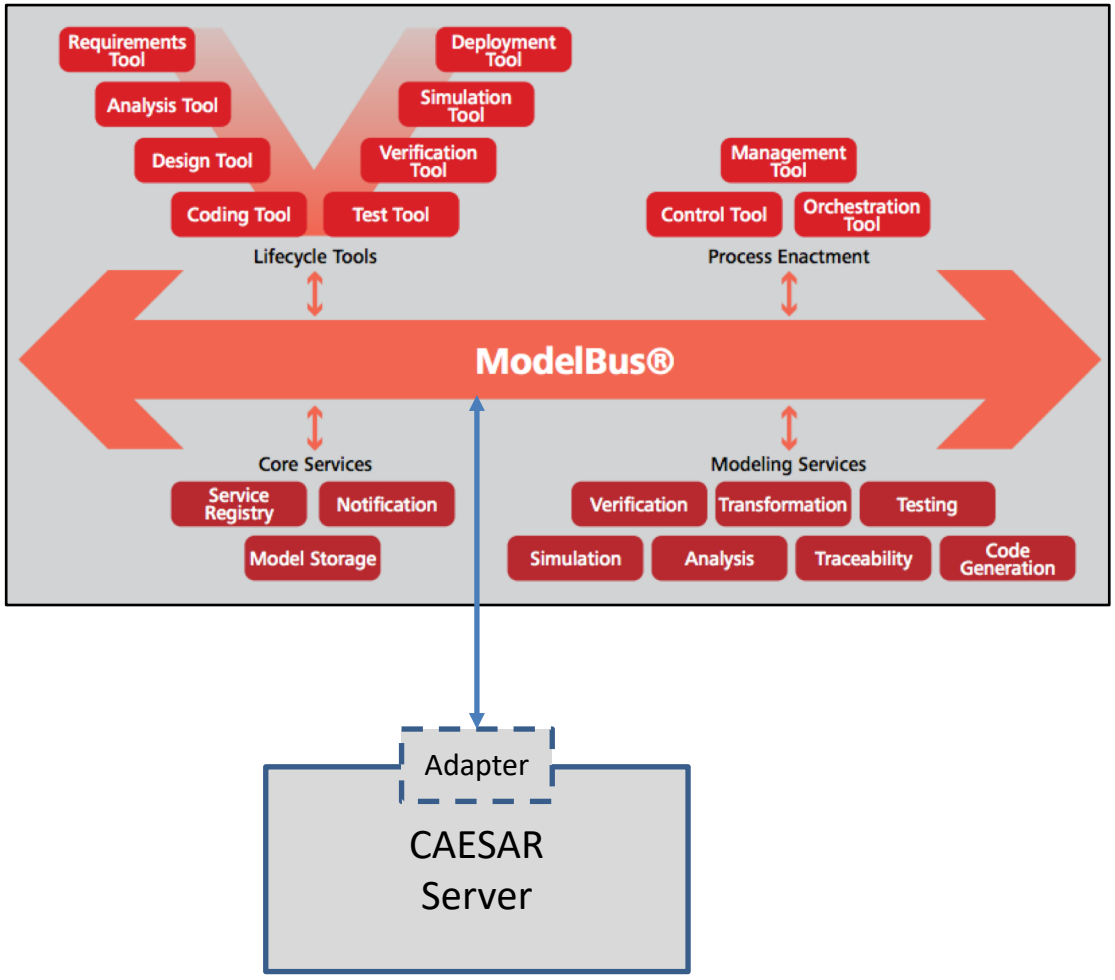
Open CAESAR vs. OSLC



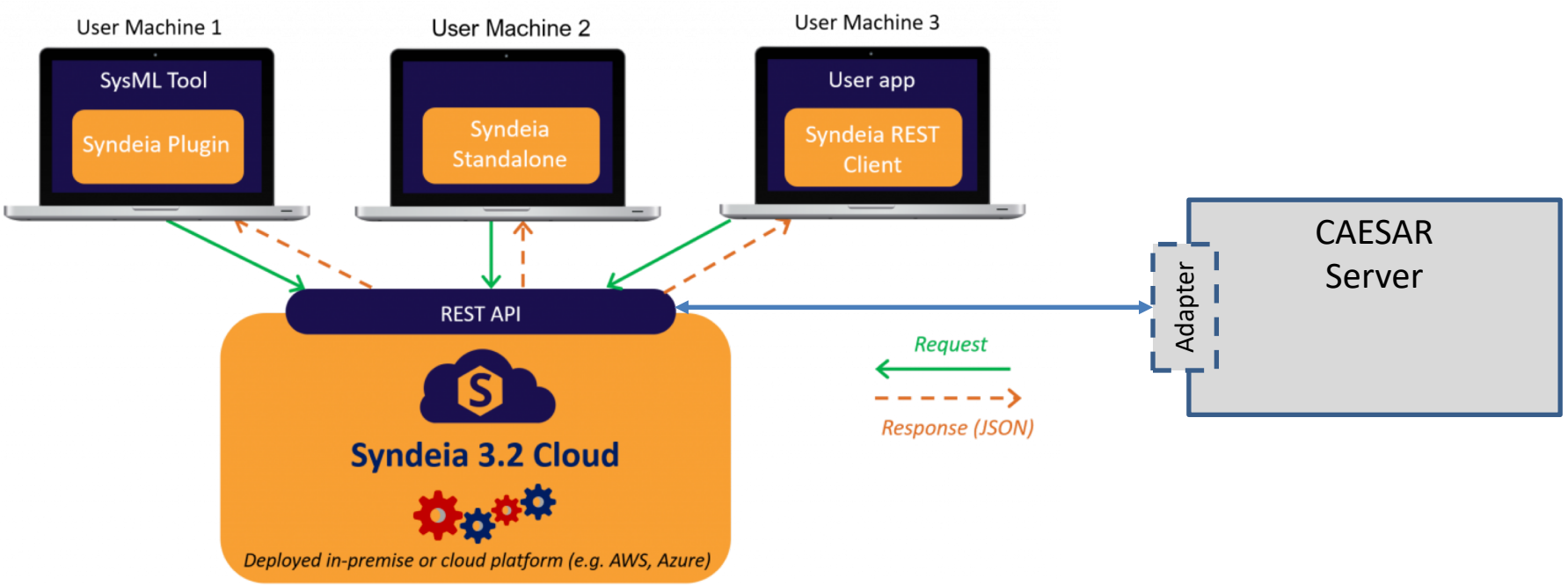
Open CAESAR vs. OpenMBEE



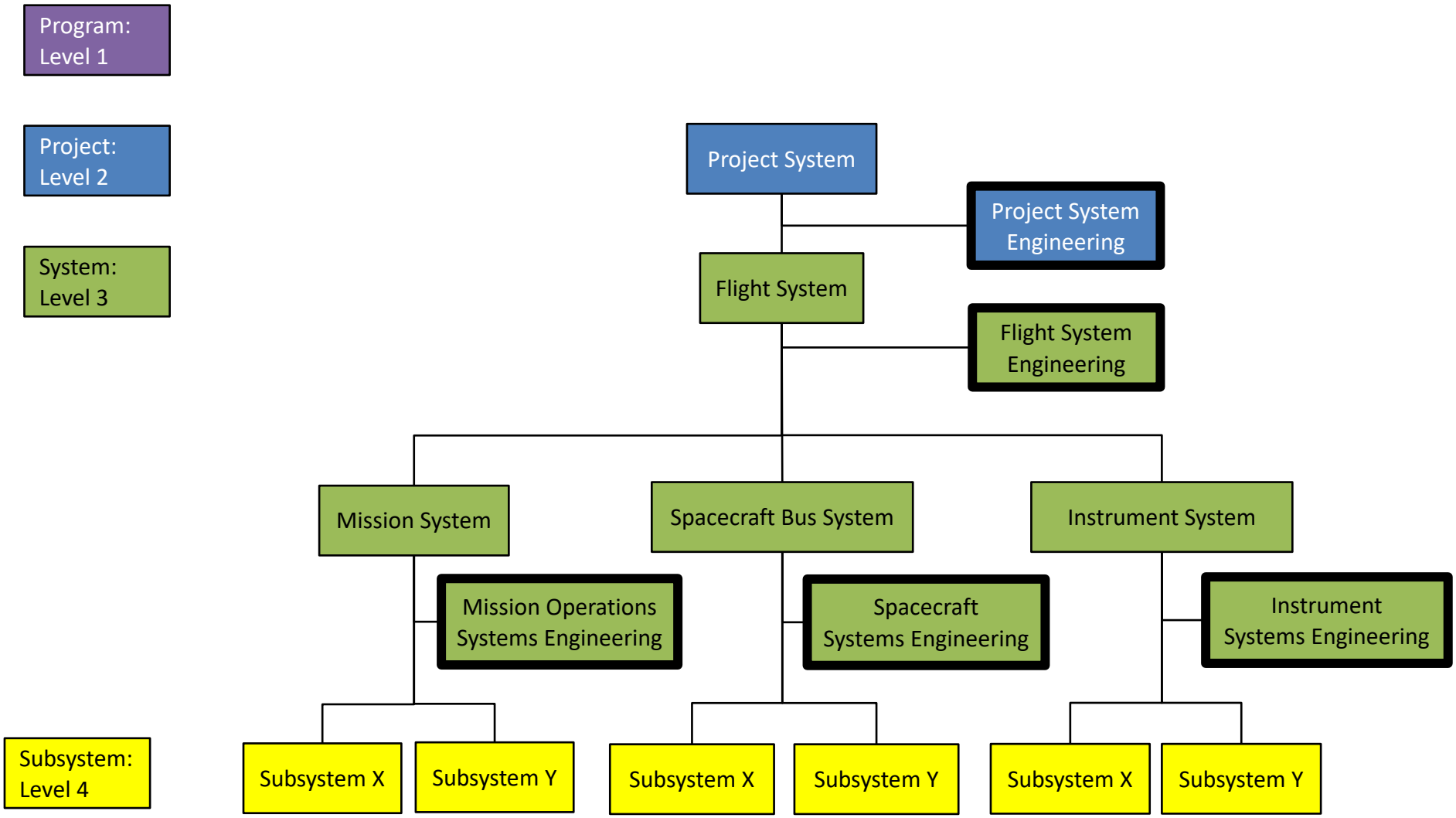
Open CAESAR vs. ModelBus



Open CAESAR vs. Syndeia



JPL Systems Engineering Domains



JPL 10 Systems Engineering Functions (Disciplines)

- Manage technical resources and performance
- Manage and control change
- Specify requirements
- Define interfaces
- Characterize and analyze design
- Perform verification and validation
- Manage tasks
- Manage risk
- Conduct reviews
- Define architecture

JPL Flight Systems Engineering Applications